

Kinematikmodell für Komponenten mehrachsiger Maschinen

Dipl.-Ing. Lee, Seok Won; Dipl.-Ing. Nguyen, Viet Duc

1 Einleitung und Zielstellung

„Kinematikmodell für Komponenten mehrachsiger Maschinen“ ist ein Bestandteil der Maschinensimulation. Zur Zeit gibt es verschiedene Softwarelösungen, die eine 5-Achsen-Werkzeugmaschinen gut simulieren kann. Beispiele sind die Software Vericut oder GIBCam Virtro usw. Trotzdem besteht der Bedarf zur Weiterentwicklung der Software oder als Alternative die komplette Neuentwicklung. Es gibt mehre Gründe dafür. Außer der Weiterentwicklung sind lizenzrechtliche Fragen, Kosten für den Kauf der Software sowie ein hoher Kommunikationsaufwand mit dem Hersteller zu beachten. Dagegen ist die komplette Eigenentwicklung mit Vorteilen wie freie Gestattung der Schnittstellen und problemlose Integration verbunden.

Mit Hilfe dreidimensionaler Simulation wurden die kinematischen Ketten von beliebigen Maschinen bzw. Maschinenelementen untersucht. Dazu wurde ein Programm erstellt, welches es ermöglicht, ausgehend von technischen Zeichnungen oder vorhandenen CAD-Daten, ein funktionales Abbild der Maschine zu erstellen und dessen einzelne Maschinenelemente entsprechend ihrer kinematischen Eigenschaften (Rotation, Translation etc.) zu verknüpfen. Außerdem kann man Werkzeuge flexibel auswählen.

Der Bewegungsablauf der Maschine soll in zwei Arten gesteuert werden. Erstens durch Jogging, das heißt, dass man die Bewegung der Maschine durch ein Dialogfenster einstellen kann. Und zweitens die Steuerung der Maschinenbewegungen durch ein NC-Programm.

Parallel dazu soll jederzeit eine detaillierte Kollisionskontrolle erfolgen, um so bereits in der Simulationsphase eventuelle Konstruktionsmängel der Maschinenelemente und Fehler in den Anlagen-Programmen aufdecken zu können.

2 Kinematische Kette

In diesem Teilabschnitt sollen das modulare Maschinenmodell mit den kinematischen Eigenschaften betrachten werden. Die Maschinenelemente können in unterschiedlichen CAD-Formaten vorliegen. In dieser Arbeit wird zunächst auf das STL-Datenformat eingegangen. Der Grund dafür ist die Einfachung und Schnelligkeit beim Lesen und Interpretieren.

Jede Maschine hat eine eigene kinematische Kette. Deshalb soll hier ein allgemeiner Algorithmus gefunden werden, damit man die kinematischen Ketten für beliebige Fälle beschreiben bzw. erzeugen kann.

Um diese Aufgabe zu lösen, muss man zunächst die mathematische Beziehung zwischen den Maschinenelementen finden. Zwischen zwei Maschinenelementen gibt als Bewegungsbeziehung drei verschiedene Fälle. Ein Element kann translatorisch, rotatorisch oder fest mit einem anderen Element verbunden sein. Und für jeden Fall muss eine mathematische Beziehung erzeugt werden. Nachdem die mathematische

Beziehung zwischen den Maschinenelementen erzeugt wurde, muss die Transformationsmatrix von i-Maschinenelement-Koordinatensystem auf das Basis-Koordinatensystem berechnet werden.

2.1 Mathematische Beziehung zwischen zwei Elementen

Es wird angenommen, dass es zwei Elemente wie in der Abbildung 1a gibt. Die Koordinaten aller Vektoren und Punkte in der Abbildung sind bekannt. Das Ziel ist die Transformationsmatrix zwischen zwei Elementen zu berechnen, damit die zwei Kontaktflächen, Vektor \vec{pn}_i und Vektor \vec{pv}_{i+1} ; Vektor \vec{pnt}_i und Vektor \vec{pvt}_{i+1} identisch sind (Abbildung 1b).

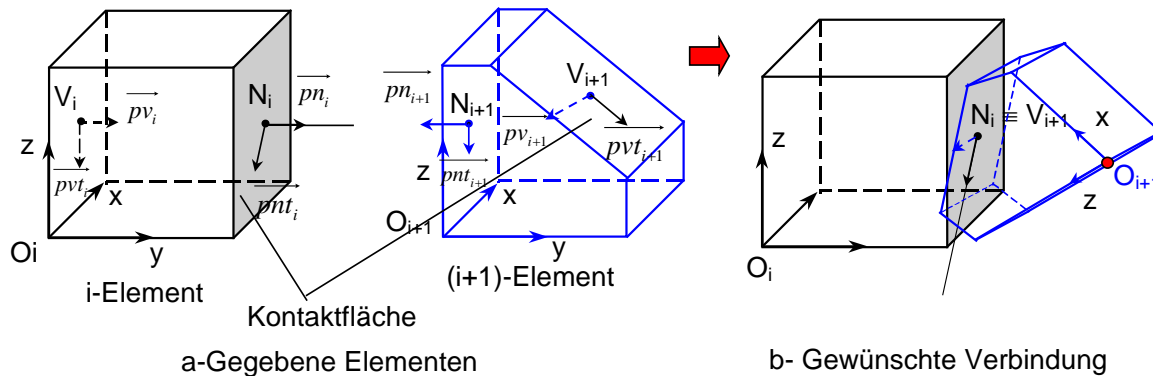


Abbildung 1: zwei einfache Elemente der Maschine [NGU-06]

Zuerst wird die Transformationsmatrix berechnet, damit die Initialisierungsfigur der Verbindung entsteht. Und zwar wie folgt: Zu erst bewegt sich der Ursprung des (i+1)-Koordinatensystems zu Punkt N_i . Dann dreht sich das (i+1)-Koordinatensystem um einen α -Winkel, damit Vektor \vec{pn}_i und Vektor \vec{pv}_{i+1} identisch sind. Danach dreht sich das (i+1)-Koordinatensystem um einen β -Winkel, damit Vektor \vec{pnt}_i und Vektor \vec{pvt}_{i+1} identisch sind. Am Ende bewegt sich das (i+1)-Koordinatensystem zu Punkt N_i , damit Punkt N_i und Punkt V_{i+1} identisch sind.

Für eine Maschine könnte es sein, dass sich das (i+1)-Element gegenüber i-Element dreht oder verschiebt oder nicht bewegt. Die Berechnung der Transformationsmatrix für jeden Fall ist nicht gleich. Bei Drehung bewegt sich zunächst die Ursprung des (i+1)-Koordinatensystems zu Punkt N_i . Dann dreht sich ein φ -Winkel um den \vec{pv}_{i+1} -Vektor. Schließlich bewegt sich das (i+1)-Koordinatensystem, damit der Punkt V_{i+1} und der Punkt N_i identisch sind. Bei Verschiebung bewegt sich (i+1)-Koordinatensystem entlang des \vec{pnt}_i -Vektors.

2.2 Transformationsmatrix auf Basiskoordinatensystem

Die Kinemattkette nach Abbildung 2 wird betrachtet. Die Transformationsmatrix des i-Koordinatensystem wird auf das Basiskoordinatensystem bezogen. Diese Transformationsmatrix berechnet sich folgendermaßen:

$$T_{ai} = T_{a1}T_{a2} \dots T_{a(i-1)}T_{ai} \quad (2.1)$$

Dabei ist T_{ai} die Transformationsmatrix zwischen (i-1)-Koordinatensystem und i-Koordinatensystem.

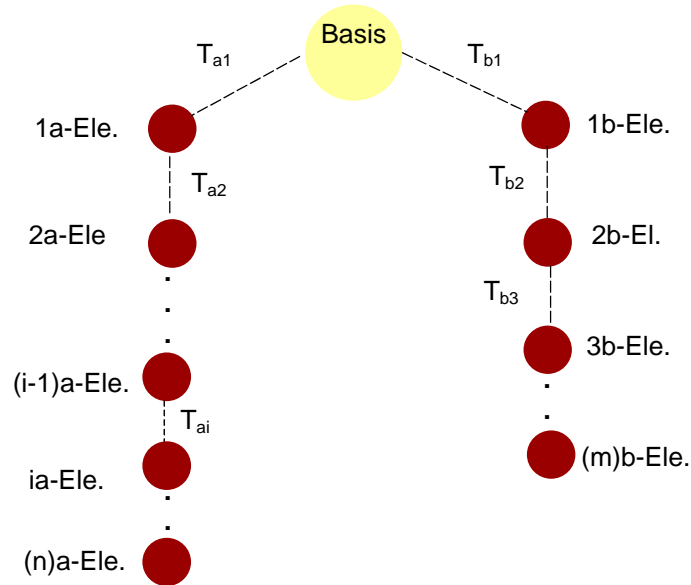


Abbildung 2: Berechnung der gesamten Transformationsmatrix [NGU-06]

3 Simulation durch NC-Programm

Für Simulation durch NC-Programm wird zuerst ein NC-Programm eingelesen. Dann werden die Zwischenpunkte berechnet. Abhängig von Rückwärtssimulation oder Vorwärtssimulation wird der Simulationsvorgang rückwärts oder vorwärts durchgeführt (Abbildung 3).

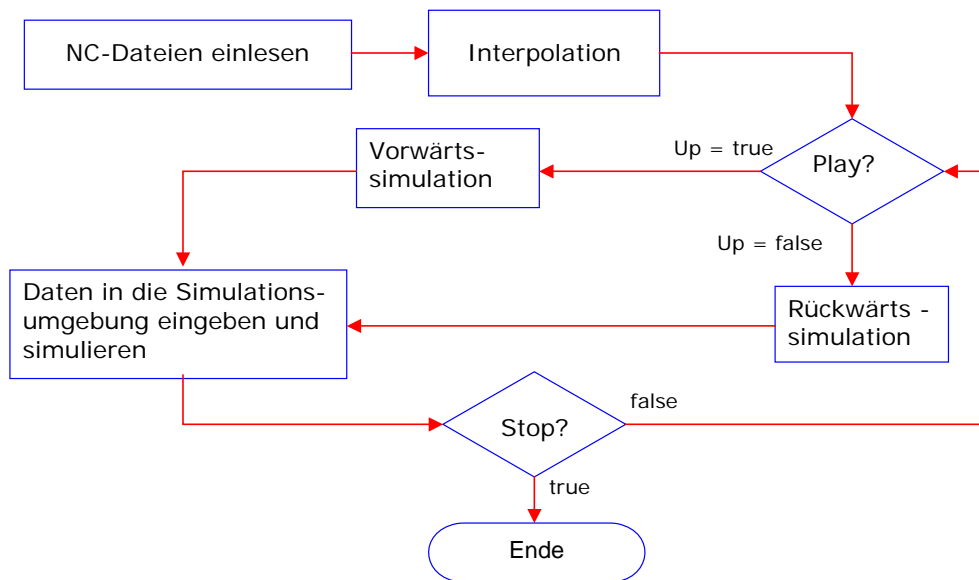


Abbildung 3: Prozessablauf für Simulation durch NC-Programm [NGU-06]

4 Kollisionserkennung

Bei Kollisionserkennung wird zuerst eine Hierarchie von Hüllkörpern erzeugt, d.h. es gibt ein Bounding Volume (BV) für das gesamte Objekt, ein BV für jede Hälfte des Objekts, für jedes Viertel, usw. Anschließend wird bei Erkennen einer Kollision die Berechnung zwischen den Hüllkörpern durchgeführt. Wenn keine Kollision auftritt,

wird der Prozess beendet. Im anderem Fall geht man eine Stufe tiefer in der Hierarchie und führt den Test nochmals durch (Abbildung 4).

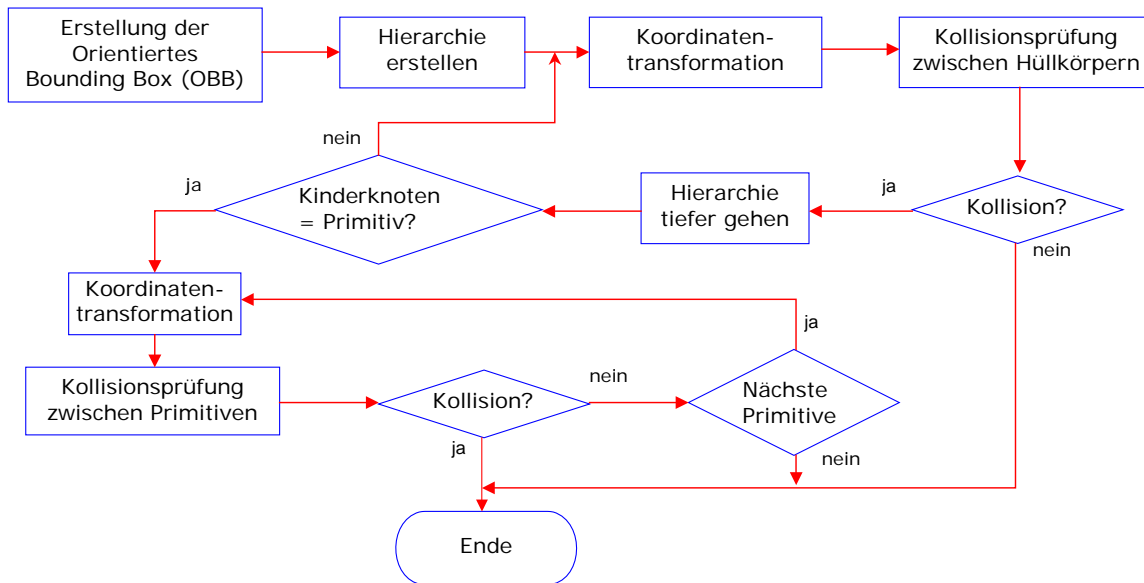
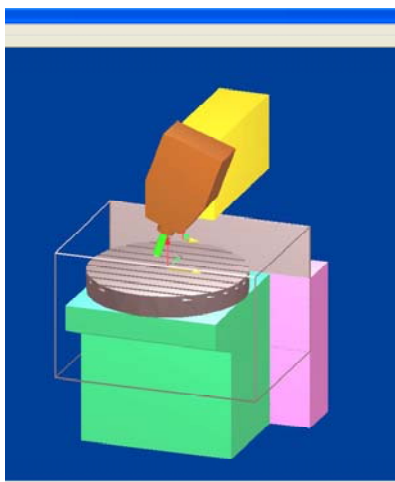


Abbildung 4: Prozessablauf für Kollisionserkennung [NGU-06]

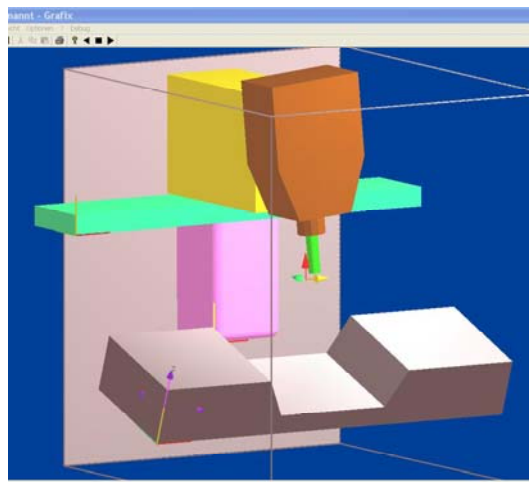
Hier wird eine so genannte „Oriented Bounding Box“ als Hüllkörper angewendet. Und für die Kollisionserkennung zwischen Hüllkörpern wird der sogenannten „Separating Axis Theorem“ [GOT-96] benutzt. Dafür werden die Objekte auf eine separierenden Achse projiziert und dann wird die Projektion ausgewertet. Der Algorithmus von O. Devillers und P. Guigue [DEV-02] wird für Kollisionserkennung zwischen zwei Dreiecken angewendet. Die Hauptidee ist die Auswertung der speziellen Determinante.

5 Ergebnisse

Die Abbildung 5 zeigt zwei kinematische Ketten. Auf der linken Seite ist die kinematische Kette der Maschine MAHO800C. Die rechte Seite zeigt eine Kombination, die B-Teil auf dem Kopf und A-Teil auf dem Tisch haben.



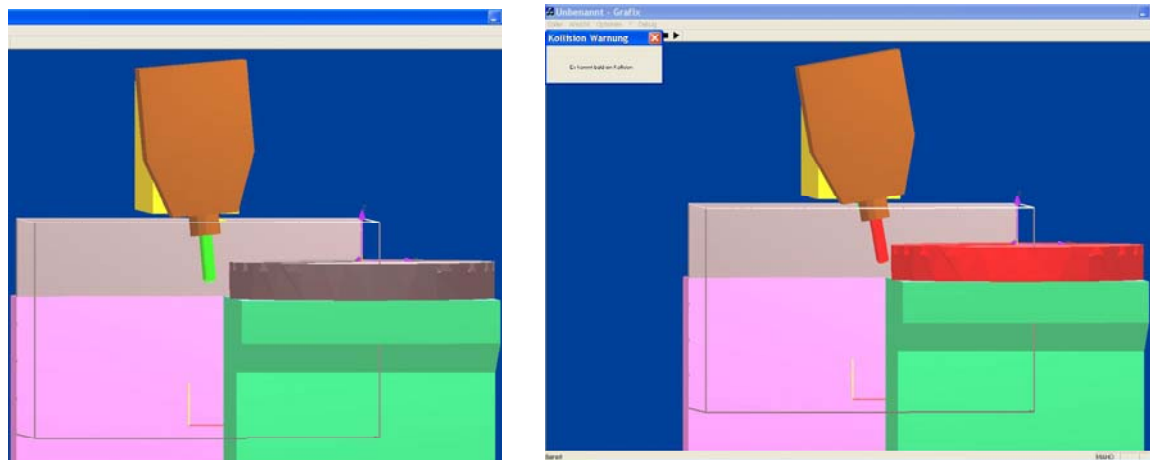
a-Schwenkkopf-Drehtisch



b-Tisch A – Kopf B

Abbildung 5: Beispiel für kinematische Kette [NGU-06]

Abbildung 6 zeigt die Kollisionserkennung bei manuellen Handhabung. Hier gibt es eine Kollision zwischen Werkzeug und Drehtisch, dann wird die Farbe des Werkzeuges und Drehtisches rot geändert.



- a- Vor der Kollision
- b- Kollisionserkennung

Abbildung 6: Kollisionserkennung bei manueller Handhabung [NGU-06]

Die Kollisionserkennung bei der durch ein NC-Programm gesteuerten Handhabung wird durch Abbildung 7 dargestellt. Gibt es eine Kollision, dann kommt ein Information über den NC-Satz, in dem eine Kollision auftritt.

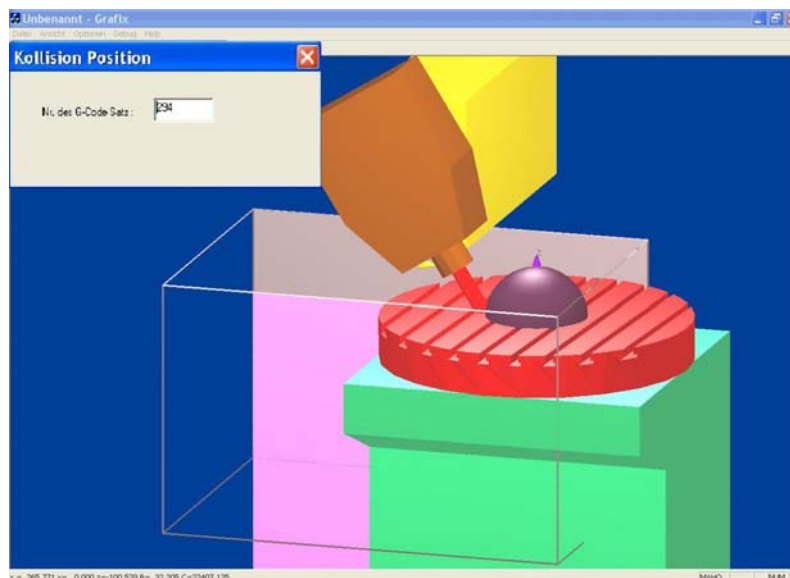


Abbildung 7: Kollisionserkennung bei der durch ein NC-Programm gesteuerten Handhabung [NGU-06]

6 Zusammenfassung und Ausblick

Mit der Arbeit wurde eine Software entwickelt, die folgende Fähigkeiten hat:

- Simulation von beliebigen Kinemattketten für 5-Achsen-Werkzeugmaschinen mit beliebigem Werkzeug (Kugelfräser, Schafffräser und Torusfräser).
- Steuerung der Werkzeugmaschinenbewegungen mittels NC-Programm.

- Kollisionserkennung zwischen je zwei Maschinenelementen und Erzeugung einer Meldung, wenn eine Kollision auftritt.

Das erzeugte Programm geht von nachfolgenden Bedingungen aus:

- CAD-Daten für Maschinenkomponenten müssen im Format von STL-Dateien vorliegen.
- Die INI-Dateien müssen vorher erzeugt worden sein.
- Das NC-File muss nach DIN66025 geeignet sein.

Mit den vorliegenden Bedingungen deckt das Programm nur einen kleinen Bereich in der Fertigung ab. Jedoch, dieses Programm kann erweitert werden:

- Die Erweiterung kann für andere NC-Steuerungen (Fanuc, Heidenhai, usw.) vorgenommen werden.
- Die Erweiterung kann für die Kollision zwischen anderen Maschinenkomponenten (z.B.: Spannmittel und Werkzeug, Werkzeug-Werkstück, usw.) erfolgen.
- Außerdem kann die Erweiterung auch für andere Simulationen (z.B.: Abtragsimulation, Thermische Prozesse, usw.) erfolgen.
- Die Erweiterung kann für andere Datenformate (z. B. STEP-NC usw.) vorgenommen werden.

Literatur

- [GOT-96] Gottschalk S., Lin M. C., Manocha D.: OBBTree: A Hierarchical Structure for Rapid Interference Detection ; *Computer Graphics. Proc. ACM SIGGRAPH*, 30:171.180, 1996.
- [DEV-02] Devillers, O.; Guige, P. :*Faster Triangle-Triangle Intersection Tests*. Institut National de Recherche en Informatique en Automatique. Juni 2002.
- [NGU-06] Nguyen, V. D.: Kinematikmodell für Komponenten mehrachsiger Maschinen; Institut für Produktionstechnik Arbeitsgruppe Produktionsautomatisierung, Zerspanungs- und Abtragtechnik. Masterarbeit TU Dresden 2006.